

EISTI



Ecole  
Internationale  
des Sciences  
du Traitement  
de l'Information

SIMULATION  
DE VARIABLES ALEATOIRES  
ET  
PROCESSUS STOCHASTIQUES

# SOMMAIRE

<b>1</b>	<b>Simulation de lois de probabilité .....</b>	<b>3</b>
1.1	Un petit test sur le générateur de nombres « aléatoires » .....	3
1.2	Simulation d'expériences .....	4
1.3	Simulation de lois .....	5
1.3.1	Schéma d'urnes pour lois discrètes.....	5
1.3.2	Méthode d'inversion pour lois discrètes .....	6
1.3.3	Méthode d'inversion pour lois continues .....	8
1.3.4	Simulation de la loi normale par le TCL .....	8
1.3.5	Simulation de la loi normale par le méthode de Box-Muller.....	9
<b>2</b>	<b>Processus stochastiques .....</b>	<b>10</b>
2.1	Définitions et propriétés .....	10
2.1.1	Trajectoires et états d'un processus stochastique.....	10
2.1.2	Propriétés d'un processus.....	11
2.2	Mouvement brownien .....	12
2.3.1	Définition.....	12
2.3.2	Simulation : première méthode.....	12
2.3.3	Simulation : deuxième méthode .....	13
2.3	Bruit blanc .....	13
2.4	Chaînes de Markov .....	16
2.5	Processus de Poisson ??? .....	16
<b>3</b>	<b>Annexe .....</b>	<b>16</b>
3.1	Démonstration pour les tests de bruits blancs .....	18

## REFERENCES

- Lambertion Damien et Lapeyre Bernard. « Introduction au calcul stochastique appliqué à la finance », Ed. Ellipse
- Millet Annie. « Méthodes de Monte Carlo ». Support de cours Master 2, universités Paris 7 et Paris 1.

# 1 Simulation de lois de probabilité

## 1.1 Un petit test sur le générateur de nombres « aléatoires »

```
Quitter et rouvrir Scilab
Générer cinq nombres aléatoires et noter le vecteur obtenu
Quitter et rouvrir Scilab
Générer à nouveau cinq nombres aléatoires et comparer les deux vecteurs
```

Tous les langages disposent d'un générateur pseudo-aléatoire, appelé **rand** dans Scilab. C'est une fonction qui retourne un nombre 'au hasard' entre 0 et 1 et dont on admet qu'elle vérifie les postulats suivants :

1.  $0 \leq a < b \leq 1$ ,  $\text{Proba}(\text{rand} \in ]a, b]) = b - a$
2. Les appels successifs de **rand** sont des variables aléatoires indépendantes

### Conséquences

- $\forall k \in \mathbb{N}^*$ , soient  $(R_1, \dots, R_k)$   $k$  appels successifs de **rand**. Pour tout rectangle  $D = ]a_1, b_1] \times \dots \times ]a_k, b_k]$ ,  $0 \leq a_i < b_i \leq 1$   $i = 1, \dots, k$   
 $\text{Proba}[(R_1, \dots, R_k) \in D] = (b_1 - a_1) \dots (b_k - a_k)$ .

Par exemple pour  $k=2$ , la probabilité qu'un point dont les coordonnées sont des appels successifs de **rand** tombe dans un rectangle est la surface de ce rectangle.

La probabilité pour que **rand** tombe sur une valeur particulière est nulle

$$\forall a \in [0, 1], \text{Proba}[\text{rand} = a] = 0,$$
$$\Rightarrow \text{Proba}(\text{rand} \in [a, b]) = \text{Proba}(\text{rand} \in ]a, b]) = \text{Proba}(\text{rand} \in ]a, b[).$$

Il y a un paradoxe à admettre à la fois que **rand** peut prendre une infinité de valeurs distinctes et ne prend jamais aucune valeur particulière. En pratique la situation est légèrement différente. Premièrement, pour l'ordinateur, il n'existe qu'un nombre fini de valeurs. Les valeurs de **rand** sont toujours calculées à partir d'entiers répartis dans  $\{0, 1, \dots, M\}$ , où  $M$  est de l'ordre de  $10^8$  au moins. Pour retourner un réel entre 0 et 1, il suffit de diviser par  $M$ . En pratique, seul un nombre fini de valeurs peuvent être atteintes. Ceci contredit les postulats de définition de **rand** mais ne constitue pas un inconvénient majeur dans la mesure où  $M$  est très grand. Deuxièmement la valeur retournée par un générateur est une fonction de la (ou les) précédente (s) (d'où générateur pseudo-aléatoire). En fait on a à faire à une suite récurrente sur un ensemble fini donc périodique. On considère ici que la suite est aléatoire dans la mesure où la période est très grande. Il existe différents types d'algorithmes pour générer des suites de nombres pseudo-aléatoire qui ont tous besoin d'une 'graine' pour initialiser la première valeur de la suite. **Il est donc indispensable de faire figurer une instruction de randomisation (une seule fois) en début de session.** Dans Scilab, cette instruction est

```
rand("seed", n) .
```

## 1.2 Simulation d'expériences

En admettant les postulats sur le générateur aléatoire, il est possible de simuler des suites **d'expériences aléatoires indépendantes** et donc de calculer de façon approchée des probabilités par l'application de la loi des grands nombres :

Rappel : soit  $X_1, \dots, X_n$  une suite de variables aléatoires indépendantes de même loi tel que  $E(X_i) = \mu$  alors  $(X_1 + \dots + X_n)/n$  converge en probabilité vers  $\mu$ .

```
n_A ← 0
Répéter n fois
  Expérience
  Si A réalisé alors n_A ← n_A + 1
  Fin si
Fin répéter
f_A ← n_A / n
```

La variable  $f_A$  est la fréquence de l'événement A au cours des n expériences, elle donne donc une valeur approchée de la probabilité pour que A se réalise.

Exemple : Lancer d'un dé

```
n_A ← 0
Répéter n fois
  D ← Rand({1, ..., 6})
  Si D ≥ 4 alors n_A ← n_A + 1
  Fin si
Fin répéter
f_A ← n_A / n
```

En sortie de cet algorithme,  $f_A$  donne une valeur approchée de la probabilité d'obtenir un nombre supérieur ou égal à 4. Ce nombre sera d'autant plus proche de 0.5 que n est grand. Plus le nombre d'expériences est grand, plus précis est le résultat. **Il est donc essentiel qu'une boucle de simulation soit la plus rapide possible. Il faut en éliminer toutes les opérations coûteuses et inutiles et faire le moins de boucles possibles.** Pour comparer l'efficacité des algorithmes, on peut utiliser les commandes **tic** (début) et **toc** (fin) qui permettent de mesurer le temps CPU écoulé. Pour vous rendre compte de la nécessité d'optimiser vos programmes, taper les instructions suivantes :

```
A = [];
tic()
for i=1:1000,
  for j=1:1000,
    A(i,j)=rand();
  end;
end;
toc()
tic()
A=rand(1000,1000);
toc()
```

Simulation 1 : Lancer d'un dé

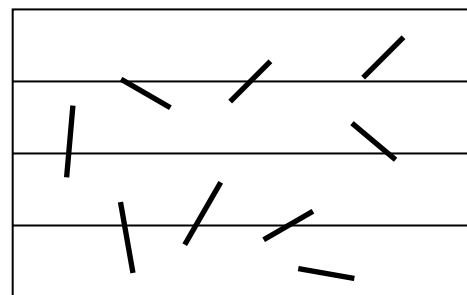
| Construire un échantillon formé de 10 000 lancers de dés

- Tracer le diagramme en bâtons des fréquences de l'échantillon
- Donner sa moyenne et sa variance
- Calculer une valeur approchée de la probabilité de l'évènement : «obtenir les faces 1 ou 6»
- Comparer avec les résultats théoriques

### Simulation 2 : L'aiguille de Buffon

On lance au hasard une aiguille sur un parquet. On supposera pour simplifier que la longueur de l'aiguille est égale à la largeur d'une lame de parquet. Le problème consiste à calculer la probabilité pour que l'aiguille tombe à cheval sur 2 lames de parquet.

Une concrétisation de cette expérience se trouve au palais de la découverte : les "lames de parquet" sont métalliques, l'aiguille est retenue par un électro-aimant et tombe quand le visiteur appuie sur un interrupteur. Si elle tombe à cheval sur deux lames, il y a contact et un compteur est incrémenté. On peut donc calculer la fréquence expérimentale. Celle-ci est remarquablement proche de  $2/\pi$  (des millions de visiteurs ont appuyé sur le bouton...). On a donc un moyen "expérimental" de calculer  $\pi$ .



Pour le modèle mathématique, on admet les hypothèses suivantes :

- La position du milieu ( $m$ ) de l'aiguille est un réel au hasard entre 0 et  $1/2$ .
- L'angle ( $\theta$ ) de l'aiguille avec l'axe vertical est un réel au hasard entre 0 et  $\pi/2$ .
- Les choix de ces deux quantités sont des expériences indépendantes.

Comme conséquence du modèle mathématique, on peut démontrer que la probabilité cherchée vaut  $2/\pi$ .

- Simuler le lancer d'une aiguille
- Tester si l'aiguille coupe une lame de plancher :  $\cos(\theta) \geq 1 - 2m$
- Donner une valeur approchée de  $\pi$

On remarquera que le programme utilise la valeur de  $\pi$  pour au final fournir une valeur approchée de  $\pi$ . Il n'a donc qu'un intérêt méthodologique.

## 1.3 Simulation de lois

Voici quelques méthodes pour simuler des lois classiques discrètes ou continues.

### 1.3.1 Schéma d'urnes pour lois discrètes

Certaines lois discrètes peuvent se ramener à un schéma d'urnes (en référence au tirage de boules noires et boules blanches). Prenons l'exemple de la loi géométrique. Le problème est d'observer une suite de répétitions indépendantes d'une même expérience. On s'intéresse au moment où **l'évènement A se produit pour la première fois**. On suppose que la probabilité  $p$  de A est strictement positive. Notons X le rang de l'expérience au cours de laquelle A se produit pour la première fois. X est une variable

aléatoire, c'est un nombre qui dépend de l'expérience aléatoire « répéter indépendamment jusqu'à ce que A soit réalisé ». L'ensemble des valeurs possibles pour X est {1,2,...}. Pour tout k>0, on a

$$\text{Proba}[X = k] = \text{Proba}[\underbrace{\overline{A} \dots \overline{A}}_{k-1} A] = \underbrace{(1-p) \dots (1-p)}_{k-1} p = p(1-p)^{k-1}.$$

La variable X suit donc une loi géométrique de paramètre p.

Algorithme

```
X ← 0
Répéter
X ← X+1
Jusqu'à (rand<p)
```

Simulation 3a : Loi géométrique

Simuler une loi géométrique  
Représenter graphiquement sa distribution empirique  
Calculer sa moyenne et sa variance – Comparer avec les résultats théoriques

Simulation 3b : Loi binomiale

Ecrire l'algorithme pour la simulation d'une loi binomiale b(n,p)  
Simuler la loi binomiale  
Représenter graphiquement sa distribution empirique  
Calculer sa moyenne et sa variance – Comparer avec les résultats théoriques

Lois concernées : binomiale, binomiale négative, géométrique, hypergéométrique

### 1.3.2 Méthode d'inversion pour lois discrètes

Soit X une variable aléatoire à valeurs  $\{x_i, i \in \{1, \dots, n\}$  ou  $i \in \mathbb{N}$  où  $x_i < x_{i+1}\}$  telle que  $P[X=x_i]=p_i, i>0$ . La fonction de répartition correspondante est définie par

$$F(x) = \begin{cases} 0 & \text{si } x < x_1 \\ p_1 + \dots + p_i = F_i & \text{si } x_i \leq x < x_{i+1} \end{cases}$$

L'algorithme de simulation par inversion est l'algorithme naturel de choix entre plusieurs éventualités. Il suppose que l'on connaisse explicitement les  $F_i$ .

Algorithme

```
i ← 1
Choix ← rand
Tant que (choix > Fi) faire
    i ← i+1
Fin tant que
X ← xi
```

Le nombre de tests de l'algorithme vaut i avec une probabilité  $p_i$ . On a donc tout intérêt à ranger les éventualités par ordre de probabilités décroissantes.

Exemple : Simulation de la loi de Poisson

Soit  $X$  une variable aléatoire de loi de Poisson de paramètre  $\lambda$ . Il n'y a pas d'expression simple pour la fonction de répartition et l'ensemble des valeurs possibles est infini. Il faut donc dans ce cas particulier calculer les valeurs de  $F_i$  au fur et à mesure. On écrit pour cela la loi de probabilité sous forme récursive :

$$\text{Prob}[X = n] = e^{-\lambda} \frac{\lambda^n}{n!} = \frac{\lambda}{n} \text{Prob}[X = n - 1].$$

```
P ← e-λ, F ← P, X ← 0
Choix ← rand
Tant que (choix > F)
    X ← X + 1, P ← P * λ / X, F ← F + P
Fin tant que
```

Mais simuler une seule valeur n'a aucun sens. L'algorithme ci-dessus sera sans doute appelé un grand nombre de fois et recalculera à chaque fois les mêmes valeurs de la fonctions de répartition, d'où une perte de temps. Voici par exemple les 6 premières valeurs de cette fonction pour  $\lambda=1$

i	0	1	2	3	4	5
$F_i$	0,3679	0,7358	0,9193	0,9810	0,9963	0,9994

On a donc tout intérêt à mettre dans un tableau en début de programme ces 6 valeurs, quitte à calculer les suivantes le cas échéant.

```
Tableau des F[i], i=0,...,Max
Fmax ← F[Max], Pmax ← F[Max] - F[Max-1]
Choix ← rand
Si (choix ≤ Fmax)
    Alors
        X ← 0
        Tant que (choix > F[X])
            X ← X + 1
        Fin tant que
    Sinon
        X ← Max, P ← Pmax, F ← Fmax
        Tant que (choix > F)
            X ← X + 1, P ← P * λ / X, F ← F + P
        Fin tant que
    Fin si
```

#### Simulation 4a : Loi de Poisson

Simuler une loi de Poisson de paramètre  
Représenter graphiquement sa distribution empirique  
Calculer sa moyenne et sa variance – Comparer avec les résultats théoriques

#### Simulation 4b : Loi géométrique

Ecrire l'algorithme pour simuler une loi géométrique par inversion  
Simuler la loi géométrique

Représenter graphiquement sa distribution empirique  
Calculer sa moyenne et sa variance – Comparer avec les résultats théoriques  
Comparer avec la méthode par schéma d’urnes

Lois concernées : Toutes les lois discrètes. Si on l’utilise correctement, la méthode d’inversion est très performante en particulier quand un faible nombre de valeurs ont une probabilité cumulée proche de 1.

### 1.3.3 Méthode d’inversion pour lois continues

#### Proposition

Soient  $F$  une fonction de répartition et  $U$  une variable aléatoire de loi uniforme sur  $[0,1]$ . La variable aléatoire  $X=F^{-1}(U)$  a pour fonction de répartition  $F$ .

Il suffit donc de connaître l’inverse de la fonction de répartition pour pouvoir simuler la loi. Exemple de la loi exponentielle de paramètre  $\lambda$ .

$$\forall x \geq 0, F(x) = 1 - e^{-\lambda x} \text{ donc } \forall u \in [0,1], F(x) = u \Leftrightarrow x = -\log(1-u)/\lambda.$$

#### Algorithme

$$X \leftarrow -\log(\text{rand}) / \lambda \quad (\text{car } 1 - \text{rand} = \text{rand})$$

#### Simulation 5a : Loi exponentielle

Simuler une loi de exponentielle de paramètre 2  
Représenter graphiquement sa distribution empirique  
Calculer sa moyenne et sa variance – Comparer avec les résultats théoriques

#### Simulation 5b : Loi de Cauchy

La fonction de répartition de la loi de Cauchy est

$$\forall x \in \mathbb{R}, F(x) = \frac{1}{\pi} \arctan(x) + \frac{1}{2}$$

Ecrire l’algorithme puis simuler une loi de Cauchy. On note que la loi de Cauchy n’admet aucun moment fini.

#### Lois concernées : exponentielle, Weibull, Cauchy

### 1.3.4 Simulation de la loi normale par le TCL

Il est uniquement nécessaire de savoir simuler la loi normale centrée réduite puisque si  $Z$  suit une  $N(0,1)$  alors  $Y = \sigma Z + \mu$  suit une  $N(\mu, \sigma^2)$ .

Soit  $(X_n)_{n>0}$  une suite de variables aléatoires indépendantes de même loi, d’espérance  $\alpha$  et de variance finie  $\beta^2$ . Alors

$$Z_n = \frac{\sqrt{n}}{\beta} (\bar{X}_n - \alpha), \text{ où } \bar{X}_n = \frac{X_1 + \dots + X_n}{n}$$

converge en loi vers la loi normale  $N(0,1)$ .

#### Algorithme

$M \leftarrow 0$   
Répéter  $n$  fois

```
Xi ← simulation loi ( $\alpha$ ,  $\beta^2$ )  
M ← M + Xi  
Fin répéter  
Z ← (M -  $\alpha$ ) *  $\sqrt{n}/\beta$ 
```

Problème : A partir de quel rang peut-on considérer que la convergence est atteinte ? Cela dépend de la loi utilisée. Par exemple si on utilise une suite de lois uniformes sur  $[0,1]$  alors 12 itérations suffisent en général. Cela vient du fait que  $\alpha=1/2$  et  $\beta^2=1/12$  donc  $S=(X_1+\dots+X_{12})/12$  suit une loi  $N(1/2,1/12)$  et  $X=12[S/12-1/2]$  suit une  $N(0,1)$  donc pour  $n=12$  les calculs sont simplifiés car  $X=S-6$ .

Simulation 6a : Loi normale

Simuler loi normale  $N(0,1)$  en utilisant une loi uniforme (Représentation graphique, moyenne, variance)  
Simuler loi normale  $N(0,1)$  en utilisant une loi exponentielle (Représentation graphique, moyenne, variance)  
Comparer les vitesses de convergence  
Simuler une loi normale de paramètres  $\mu$  et  $\sigma^2$ .

### 1.3.5 Simulation de la loi normale par le méthode de Box-Muller

Cette méthode est basée sur la proposition suivante. Soient  $R$  un v.a. de loi exponentielle de paramètre  $1/2$  et  $\Theta$  une v.a. indépendante de  $R$  de loi uniforme sur  $[0,2\pi]$ . On note  $X = \sqrt{R} \cos \Theta$  et  $Y = \sqrt{R} \sin \Theta$ . Alors  $X$  et  $Y$  sont 2 v.a. indépendantes de loi normale  $N(0,1)$ .

Remarque : Le fait d'obtenir deux v.a. indépendantes de loi  $N(0,1)$  permet de diviser par 2 le nombre d'itérations à effectuer pour construire un échantillon de taille  $n$ .

Simulation 6b : Loi normale

Ecrire l'algorithme puis simuler la loi normale  $N(0,1)$  (Représentation graphique, moyenne, variance)  
Comparer avec la méthode précédente

## 2 Processus stochastiques

### 2.1 Définitions et propriétés

Soit  $(\Omega, \mathcal{F}, P)$  un espace probabilisé. Une fonction aléatoire est une application de  $\Omega$  dans un espace de fonctions. On s'intéresse ici aux fonctions aléatoires réelles, c'est-à-dire qu'à tout  $\omega \in \Omega$  correspond une fonction d'une variable réelle  $t$  :

$$\omega \mapsto X(t, \omega) = X_t(\omega).$$

Usuellement, l'argument  $t$  représente le temps et on parle alors de processus aléatoire ou processus stochastique.

#### Définition

Soient  $(\Omega, \mathcal{F}, P)$  un espace probabilisé,  $T$  un ensemble quelconque et  $(E, \mathcal{E})$  un ensemble mesurable où  $E \subset \mathbb{R}$  ou  $E \subset \mathbb{R}^n$ . On appelle *processus stochastique* défini sur  $\Omega$  avec  $T$  ensemble des temps et  $E$  espace des états, toute famille  $\{X_t\}_{t \in T}$  de variables aléatoires à valeurs dans  $E$ .

Les processus stochastiques fournissent des modèles pour un grand nombre de phénomènes dépendant du temps tels que la température, la pression, le débit, l'arrivée de clients dans un service ...

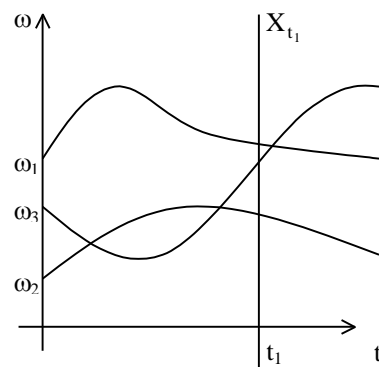
#### 2.1.1 Trajectoires et états d'un processus stochastique

Un processus est donc une application  $X$  de  $\Omega \times T$  dans  $E$ ,

$$(\omega, t) \mapsto X(\omega, t) = x(\omega, t),$$

qui recouvre deux notions :

- Si  $\omega$  est fixé, alors l'application,  $X(\cdot, \omega) : T \rightarrow E$   
 $t \mapsto X(\omega, t) = X_t(\omega)$   
est appelée *trajectoire* de  $\omega$ . Un processus peut donc être assimilé à l'ensemble de ses trajectoires lorsque  $\omega$  décrit  $\Omega$ .
- Si  $t$  est fixé, alors  $X(t, \cdot)$  définit une variable aléatoire réelle  $X_t$  appelée *état à l'instant  $t$* .



Trajectoires d'un processus aléatoire

Par la suite on distinguera deux cas :

- $T \subset \mathbb{N}$ , c'est-à-dire que le temps prend des valeurs discrètes  $t_1, t_2, \dots$ . On parle alors de processus à temps discret (ou série chronologique).
- $T = [0, t_0]$ , c'est-à-dire que le temps est continu à l'intérieur d'un intervalle. On parle alors de processus à temps continu.

### 2.1.2 Propriétés d'un processus

La plus ou moins grande régularité des trajectoires provient essentiellement de la structure des dépendances temporelles. Celle-ci peut s'exprimer de différentes façons.

Soit  $\{X_t\}_{t \in T}$  un processus aléatoire.

#### Définition

Soient  $t_1 < t_2 \leq t_3 < t_4$  où  $t_i \in T$ . On dit que le processus est à *accroissements indépendants* si les variables aléatoires,

$$X_{t_2} - X_{t_1} \text{ et } X_{t_4} - X_{t_3}$$

sont indépendantes.

#### Définition

On dit que le processus est à *accroissements stationnaires* si les variables aléatoires suivantes ont même loi,

$$X_{t_2+s} - X_{t_1+s} \text{ et } X_{t_2} - X_{t_1}.$$

#### Définition

On dit qu'un processus est *stationnaire* au sens strict si sa loi de probabilité est invariante par translation sur  $t$ , c'est-à-dire  $X_t$  et  $X_{t+h}$  ont mêmes caractéristiques.

On a alors les espérances et les variances constantes

$$E(X_t) = \mu \text{ et } \text{var}(X_t) = \sigma^2, \forall t \in \{1, \dots, T\}$$

les covariances stables par translation  $h$  du temps,

$$\text{cov}(X_t, X_{t+h}) = \sigma^2(h), \forall t \in \{1, \dots, T\}.$$

Si l'espérance est nulle, on parle de processus *centré*.

#### Remarques

- La réciproque n'est pas vérifiée. Si un processus vérifie uniquement ces trois conditions, on dit qu'il est stationnaire au sens large.
- C'est le cas de nombreux processus physique où l'origine du temps est arbitraire, mais cela n'est en général pas le cas pour des processus économiques.

#### Définition

- La fonction  $h \mapsto \sigma^2(h)$  est appelée *fonction d'auto-covariance* du processus stationnaire.
- La fonction

$$\rho: h \mapsto \frac{\sigma^2(h)}{\sigma^2}$$

est appelée *fonction d'auto-corrélation*.

On remarque que ces deux fonctions sont paires,  $\sigma^2(-h)=\sigma^2(h)$  et  $\rho(-h)=\rho(h)$ .

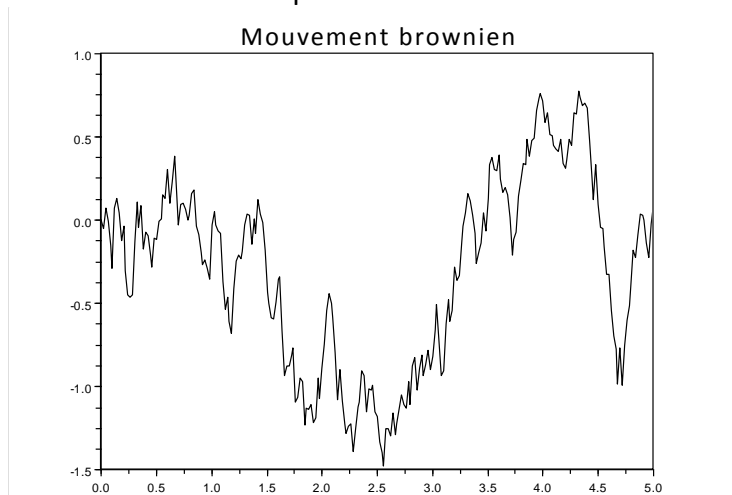
## 2.2 Mouvement brownien

### 2.3.1 Définition

On dit que  $\{B_t\}_{t \geq 0}$  est un mouvement brownien standard si et seulement si

- (1)  $B_0=0$
- (2)  $\{B_t\}_{t \geq 0}$  est un processus à accroissements indépendants
- (3)  $\forall 0 \leq s < t, B_t - B_s$  suit une loi normale  $N(0, t-s)$

Nous savons de plus que les trajectoires d'un mouvement brownien sont continues mais non différentiables p.s.



Pour simuler la trajectoire d'un mouvement brownien à temps continu, il faut d'abord discrétiser le temps. On pose  $\Delta t$  la longueur d'une période de temps et on simule le mouvement brownien aux temps  $0, \Delta t, 2\Delta t, 3\Delta t, \dots$ . On a alors  $t_n = n\Delta t$  un instant fixé.

### 2.3.2 Simulation : première méthode

La première méthode pour simuler un mouvement brownien consiste à « renormaliser » une marche aléatoire, ie une suite de variables aléatoires  $(X_k)_{k \geq 0}$  indépendantes et équidistribuées telles que  $P(X_k=1)=P(X_k=-1)=\frac{1}{2}$ . Notons  $S_m = \sum_{k=1}^m X_k$ . Alors si  $m$  est suffisamment grand, on peut approcher le mouvement brownien par

$$B_t \approx \frac{1}{\sqrt{m}} S_{[m \times t]},$$

où  $[x]$  désigne la partie entière de  $x$ .

Simulation :

- | Simuler une trajectoire d'un mouvement brownien sur l'intervalle  $[0,5]$
- | Tracer une trajectoire en faisant varier  $m=5, 10, 20, \dots$

Vérifier que pour un instant  $t$  fixé on a une loi normale  $N(0,t)$  (avec  $m=50$ )

### 2.3.3 Simulation : deuxième méthode

Cette méthode de simulation du mouvement brownien est appelé *méthode d'Euler aléatoire* et utilise le caractère gaussien des accroissements. En effet, par définition  $B_{t_{n+1}} - B_{t_n}$  suit une loi  $N(0, \Delta t)$ . Il suffit alors de poser

$$B_{t_{n+1}} - B_{t_n} = \sqrt{\Delta t} Z$$

où  $Z$  suit la loi  $N(0,1)$ . On obtient la relation,

$$B_{t_{n+1}} = B_{t_n} + \sqrt{\Delta t} Z.$$

Par récurrence, on peut alors approcher le mouvement brownien par

$$B_{t_n} \approx \sqrt{\Delta t} \sum_{k \leq n} Z_k$$

où  $(Z_k)_{k \geq 0}$  une suite de variables aléatoires indépendantes de loi normale centrée réduite.

Simulation :

Simuler une trajectoire d'un mouvement brownien sur l'intervalle  $[0,5]$   
 Vérifier que pour un instant  $t$  fixé on a une loi normale  $N(0,t)$

## 2.3 Bruit blanc

### 2.2.1 Définition

Un bruit blanc est un processus stationnaire donc la fonction d'auto-covariance est nulle, pour tout  $h$ ,  
 $h \in \{1, \dots, T\}$ .

Les bruits blancs sont les processus stationnaires les plus élémentaires. Ils ne présentent pas d'intérêt dans la prévision mais servent de base à la construction de nombreux modèles de séries chronologiques.

### Réalisation d'un bruit blanc

Il est courant d'utiliser comme bruit blanc, une suite de variables aléatoires indépendantes de loi normale centrée réduites. On parle alors de bruit blanc gaussien centré.

### 2.2.2 Test de bruit blanc

En pratique on cherche à savoir si une série d'observations  $\{x_1, x_2, \dots, x_T\}$  (signal, données chronologiques, ...) contient de l'information ou bien si elle est le fruit du hasard. Cela revient à tester si la série est un bruit blanc ou non. Il existe pour cela des tests statistiques qui utilisent la fonction l'auto-corrélation empirique comme variable de décision,

où  
 et .

- Tests individuels de bruit blanc

De façon plus formelle, il s'agit ici de tester l'auto-corrélation pour chaque retard  $h > 0$ . Sous l'hypothèse  $H_0$  que le processus  $\{x_t\}$  est un bruit blanc alors la statistique

suit approximativement une loi normale  $N(0,1)$  pourvu que  $T$  soit suffisamment grand. Ainsi si la variable  $Z_h$  prend une valeur supérieure (en valeur absolue) à celle donnée par la table de la loi  $N(0,1)$  avec un niveau de confiance  $\alpha$ , on peut rejeter  $H_0$ , c'est-à-dire l'hypothèse de bruit blanc (cf annexe).

Pour tester un bruit blanc, il convient de tester plusieurs valeurs de retard  $h$ . Cette technique présente un inconvénient majeur concernant les probabilités d'acceptation ou de rejeter de l'hypothèse de bruit blanc. En effet, pour chaque test, il y a une probabilité de  $1-\alpha$  d'accepter à raison l'hypothèse d'une absence d'auto-corrélation de retard  $h$ . Les auto-corrélation étant indépendantes, on en déduit une probabilité  $1-(1-\alpha)^H$  de rejeter l'hypothèse de bruit blanc alors qu'elle est exacte si on teste les  $H$  premières auto-corrélations. Par exemple, si on prend  $\alpha=5\%$ , il y a 5% de chance de rejeter à tort l'absence d'auto-corrélation pour chaque test. Si on teste  $H=20$  retards, il y a alors 64% de chance de rejeter à tort l'hypothèse de bruit blanc. Pour pallier ce problème, à un seuil de 5%, on rejettera l'hypothèse de bruit blanc si plus de 5% des tests individuels ont rejeté l'hypothèse. Par exemple si on prend  $\alpha=5\%$  et  $H=20$ , on rejettera l'hypothèse de bruit blanc si au moins  $20 \times 0.05 = 1$  test individuel aboutit au rejet.

- Test global de bruit blanc

Afin de ne pas effectuer des tests sur plusieurs retards, il est possible d'effectuer un test global (test du Portemanteau). Sous l'hypothèse  $H_0$  que le processus  $\{x_t\}$  est un bruit blanc alors la statistique

suit approximativement une loi du  $\chi^2$  à  $H$  degrés de liberté pourvu que  $T$  soit suffisamment grand. Ainsi si la variable  $Z$  prend une valeur supérieure à celle donnée par la table de la loi du  $\chi^2$  avec un niveau de confiance  $\alpha$ , on peut rejeter  $H_0$ , c'est-à-dire l'hypothèse de bruit blanc (cf annexe).

Cependant, ce test manque de puissance et amène à accepter un bruit blanc alors qu'il peut y avoir une auto-corrélation important pour l'un des retards.

### Méthodologie

Afin de pallier aux inconvénients des deux tests précédents, il est recommandé d'opérer de la façon suivante :

- Effectuer les tests individuels pour tous les retard de 1 à  $H$  ( $H$  pas trop grand devant  $T$ ) en s'intéressant plus particulièrement aux retards suspects ( $h=1,2,4$ , et 12).
- Effecteur le test global et considérer en même temps le nombre de rejets des tests individuels.

### Quelques commandes SAS

Cours de Aragon Y. :

[http://w3.univ-tlse1.fr/GREMAQ/Statistique/Yvesweb/docs/IUP\\_st\\_cours.pdf](http://w3.univ-tlse1.fr/GREMAQ/Statistique/Yvesweb/docs/IUP_st_cours.pdf)

Le test du portemanteau est effectué pour des sommes de retards de 6 à 24. Chi-square correspond à la valeur de Z calculée sur l'échantillon, DF le degré de liberté de la loi du chi-deux (nombre de retards) et  $\Pr\{Z > \text{ChiSq}\} = P(Z > \text{ChiSq}) = \alpha$

Quelques commandes R

```
serie=read.table(file="nom.txt",header=TRUE)
# lecture d'un tableau de données. Header= TRUE si les noms sont en entête dans le
fichier, header=FALSE sinon
plot(as.matrix(serie))
# Graphe de la série
acf(serie)
# Trace la fonction autocorrélation (cf dessin). La commande « help(acf) » permet de
voir les différentes options
# Les droites horizontales bleues donnent les seuils du test d'hypothèses pour  $\alpha=5\%$ , c-
a-d  $-1.96/\sqrt{T}$  et  $1.96/\sqrt{T}$  (cf annexe)
```

```
Box.test(serie,lag=20)
# Test portemanteaux
# X-squared est la valeur de Z calculée sur l'échantillon, df est le degré de liberté de la
loi du chi-deux (ie le nombre de retards dans la somme), p-value correspond à  $P(Z > X-
squared) = \alpha$ .
```

Box-Pierce test

```
data: serie
X-squared = 21.0609, df = 20, p-value = 0.3936
```

- Test de nullité

Les tests précédents reposent sur la fonction d'auto-corrélation et ne permettent pas de savoir si le bruit blanc est centré ou non. Pour cela, il faut faire un test de nullité. Sous l'hypothèse  $H_0$  que le bruit blanc  $\{x_t\}$  est centré alors la statistique

suit approximativement une loi normale  $N(0,1)$  pourvu que T soit suffisamment grand. Ainsi si la variable prend une valeur supérieure (en valeur absolue) à celle donnée par la table de la loi  $N(0,1)$  avec un niveau de confiance  $\alpha$ , on peut rejeter  $H_0$ , c'est-à-dire l'hypothèse que le bruit blanc est centré (cf annexe).

Exercice

Simuler un bruit blanc gaussien centré et effectuer les tests.

Les séries serie0.txt, serie1.txt, serie2.txt sont-elles des bruits blancs?

Si oui sont-elles des bruits blancs centrés ?

## 2.4 Chaînes de Markov

Définition

Simulation

## 2.5 Processus de Poisson ???

# 3 Annexe

Outils statistiques pour Scilab	
<code>rand("seed", [, n])</code>	Initialisation du générateur
<code>rand</code>	Nombre aléatoire de un loi uniforme sur [0,1]
<code>rand(m, n)</code>	Matrice $m \times n$ à coefficients i.i.d. de loi U[0,1]
<code>mean</code>	Moyenne
<code>median</code>	Médiane
<code>variance</code>	Variance empirique sans biais
<code>mvvacov</code>	Matrice de variance-covariance
<code>covar</code>	Covariance empirique de deux variables
<code>plot2d3</code>	Diagramme en bâtons (v.a. discrettes)
<code>histplot</code>	Histogramme (v.a. continues)

Des fonctions sont pré-définies dans Scilab pour simuler les lois usuelles

Simulations de lois usuelles	
<code>grand(m,n,"bin",N,p)</code>	Loi binomiale de paramètres N et p
<code>grand(m,n,"chi",n)</code>	Loi du chi-deux à n degrés de liberté
<code>grand(m,n,"def")</code>	Loi uniforme sur [0,1]
<code>grand(m,n,"exp",A)</code>	Loi exponentielle de paramètre A
<code>grand(m,n,"f",M,N)</code>	Loi de Fisher de paramètres M et N
<code>grand(m,n,"poi",L)</code>	Loi Poisson de paramètre L
<code>grand(m,n,"nor",M,S)</code>	Loi normale de paramètres M et S (écart-type)
<code>grand(n,"mn",M,S)</code>	Loi normale multidimensionnelle
<code>grand(n,"markov",P,x0)</code>	Chaîne de Markov de matrice de transition P et loi initiale $x_0$
<code>grand(m,n,"lgi")</code>	Loi uniforme sur $\{0, \dots, 2^{31}\}$
<code>grand(m,n,"uin",A,B)</code>	Entiers uniformes entre A et B
<code>grand(m,n,"unf",A,B)</code>	Réels uniformes entre A et B
<code>grand(n,"prm",V)</code>	Permutations aléatoires de V

Scilab propose des cfd à partir desquelles on retrouve la fonction de répartition, la densité et la fonction quantile.

Cumulative density fonctions	
cdfbin	cfd de la binomiale
cdfchi	cfd de la loi du chi-deux
cdff	cfd de la loi de Fisher
cdfnor	cfd de la loi normale
cdfpoi	cfd de la loi de Poisson
cdft	cfd de la loi de Student

Exemple : Soit Z de loi normale  $N(0,1)$ .

Fonction de répartition :  $P=F(Z \geq 1.64)$  et  $Q=F(Z \leq 1.64)$

$$[P,Q]=cdfnor("PQ",1.64,0,1) \Rightarrow P=0.05 \quad Q=0.95$$

Inverse de la fonction de répartition :  $F(Z \leq X)=0.05 \Leftrightarrow F(Z \geq X)=0.95 \Rightarrow X=1.64$  (table)

$$[X]=cdfnor("X",0,1,0.95,0.05) \Rightarrow 1.64$$

### 3.1 Démonstration pour les tests de bruits blancs

#### Tests individuels de bruit blanc

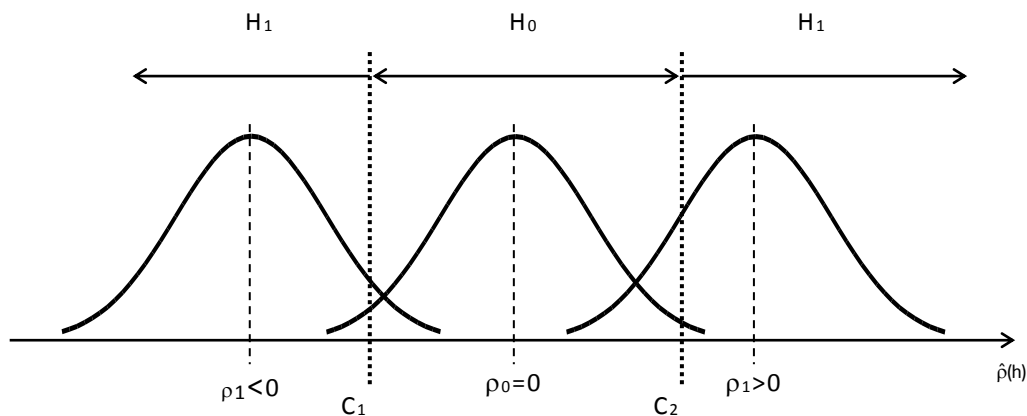
Soit  $\rho(h)$  la fonction d'auto-corrélation pour un retard  $h$ ,

Si le processus est un bruit blanc centrée alors  $\rho(h)=0$  pour tout  $h>0$ . Donc pour tout retard  $h>0$ , on effectue le test suivant.

$H_0$  = « bruit blanc centré » :  $\rho(h)=\rho_0=0$

$H_1$  = « pas un bruit blanc centré » :  $\rho(h)=\rho_1 \neq 0$

On choisit  $\hat{\rho}(h)$  l'auto-corrélation empirique comme variable de décision.



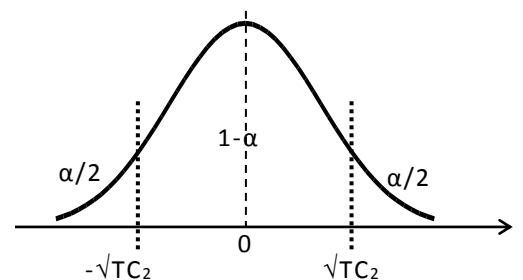
La région critique  $W$  est la région d'acceptation de  $H_1$  d'où  $W = \{\hat{\rho}(h) < C_1 \text{ ou } \hat{\rho}(h) > C_2\}$  et  $\bar{W}$  est la région d'acceptation de  $H_0$  d'où  $\bar{W} = \{C_1 \leq \hat{\rho}(h) \leq C_2\}$ .

On sait que  $\alpha = P(W | H_0 \text{ vraie})$ . Afin de simplifier le calcul de probabilité, on passe à  $1 - \alpha = P(\bar{W} | H_0 \text{ vraie})$ . Supposons  $H_0$  vraie alors on sait que  $Z_h = \sqrt{T}\hat{\rho}(h)$  suit approximativement une loi  $N(0,1)$ , d'où

$$1 - \alpha = P(C_1 \leq \hat{\rho}(h) \leq C_2) = P(C_1 \sqrt{T} \leq \sqrt{T}\hat{\rho}(h) \leq \sqrt{T}C_2)$$

On suppose que le risque est symétrique, or la loi normale centrée est aussi symétrique par rapport à  $0y$ , on a

$$\begin{cases} \sqrt{TC_1} = -\sqrt{TC_2} \\ P(Z_h > \sqrt{TC_2}) = \alpha/2 \end{cases}$$



Par exemple si  $\alpha=5\%$  alors  $\sqrt{TC_2}=1.96$ , d'où la règle de décision

- si  $|\hat{\rho}(h)| > 1.96/\sqrt{T}$  alors on rejette  $H_0$ , c-a-d on considère que ce n'est pas un bruit blanc, avec 5% de chance de se tromper
- si  $|\hat{\rho}(h)| \leq 1.96/\sqrt{T}$  alors on garde l'hypothèse d'un bruit blanc

Test global de bruit blanc

Plutôt que de tester  $\rho(h)=0$  pour tout  $h>0$ , nous allons faire un test englobant plusieurs retards  $h \in \{1, \dots, H\}$ ,

$$H_0 = \text{« bruit blanc centré »} : \sum_{h=1}^H \rho^2(h) = 0$$

$$H_1 = \text{« pas un bruit blanc centré »} : \sum_{h=1}^H \rho^2(h) > 0 \text{ (test unilatéral car positif)}$$

Si on choisit  $\sum_{h=1}^H \hat{\rho}^2(h)$  comme variable de décision, alors la région critique est

$$W = \left\{ \sum_{h=1}^H \hat{\rho}^2(h) > C \right\}. \text{ D'où } \alpha = P(W \mid H_0 \text{ vraie}).$$

Supposons  $H_0$  vraie alors on sait que  $Z = T \sum_{h=1}^H \hat{\rho}^2(h)$  suit approximativement une loi du chi-deux à  $H$  degré de liberté, d'où

$$\alpha = P\left(\sum_{h=1}^H \rho^2(h) > C\right) = P(Z > TC)$$

Par exemple si  $\alpha=5\%$  et  $H=20$ , alors  $TC=19.34$ , d'où la règle de décision

- si  $\sum_{h=1}^H \hat{\rho}^2(h) > 19.34/T$  alors on rejette  $H_0$ , c-a-d on considère que ce n'est pas un bruit blanc, avec 5% de chance de se tromper
- sinon on conserve l'hypothèse d'un bruit blanc

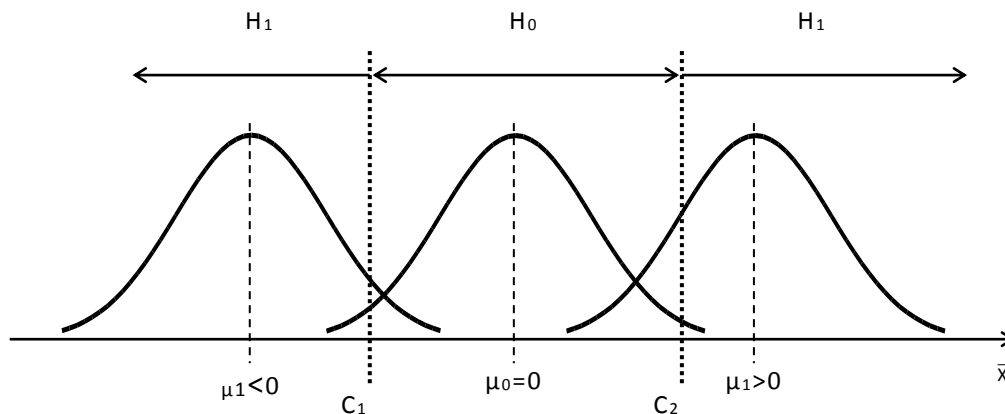
Test de nullité

On suppose que  $\{x_t\}$  est un bruit blanc (stationnaire). Donc son espérance  $\mu$  et sa variance  $\sigma^2$  sont constantes. On effectue le test suivant

$$H_0 = \text{« bruit blanc centré »} : \mu = \mu_0 = 0$$

$$H_1 = \text{« pas un bruit blanc centré »} : \mu = \mu_1 \neq 0$$

On choisit la moyenne  $\bar{X} = \frac{1}{T} \sum_{t=1}^T x_t$  comme variable de décision. Si  $T$  est suffisamment grand alors d'après le TCL, on sait que  $\bar{X}$  suit une loi  $N(\mu, \sigma^2/T)$



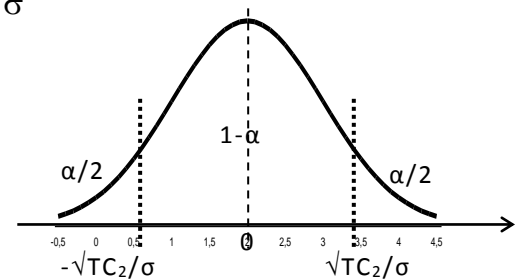
La région critique  $W$  est la région d'acceptation de  $H_1$  d'où  $W = \{\bar{X} < C_1 \text{ ou } \bar{X} > C_2\}$  et  $\bar{W}$  est la région d'acceptation de  $H_0$  d'où  $\bar{W} = \{C_1 \leq \bar{X} \leq C_2\}$ .

On sait que  $\alpha = P(W | H_0 \text{ vraie})$ . Afin de simplifier le calcul de probabilité, on passe à  $1 - \alpha = P(\bar{W} | H_0 \text{ vraie})$ . Supposons  $H_0$  vraie alors on sait que  $\bar{X}$  suit approximativement une loi  $N(0, \sigma^2/T)$ , d'où

$$1 - \alpha = P(C_1 \leq \bar{X} \leq C_2) = P\left(\frac{C_1 \sqrt{T}}{\sigma} \leq Z \leq \frac{\sqrt{TC_2}}{\sigma}\right)$$

où  $Z$  suit une  $N(0,1)$ . On suppose que le risque est symétrique, or la loi normale centrée est aussi symétrique par rapport à  $0y$ , on a

$$\begin{cases} \frac{\sqrt{TC_1}}{\sigma} = -\frac{\sqrt{TC_2}}{\sigma} \\ P(Z > \frac{\sqrt{TC_2}}{\sigma}) = \alpha/2 \end{cases}$$



Par exemple si  $\alpha = 5\%$  alors  $\sqrt{TC_2}/\sigma = 1.96$ , d'où la règle de décision

- si  $|\bar{x}| > 1.96\sigma/\sqrt{T}$  alors on rejette  $H_0$ , c-a-d on considère que ce n'est pas un bruit blanc, avec 5% de chance de se tromper
- si  $|\bar{x}| \leq 1.96\sigma/\sqrt{T}$  alors on garde l'hypothèse d'un bruit blanc